**Tiny Transformers: Enabling Transformer Execution on Low-Power IoT Endnodes**

Moritz Scherer¹, Alessio Burrello², Marcello Zanghieri², Luca Benini¹,², and Francesco Conti²

¹ETH Zurich, ²University of Bologna

---

**Self-Attention**

Transformers are SoA in many fields

- Computer Vision, Audio, NLP, many more

Self-attention is key layer in transformers

- Linear layers + many parameters
- Non-trivial data dependencies

Multithead Self-Attention

- Linear Layer
- Linear Layer
- Linear Layer
- Concet & Transpose
- Matmul
- Softmax
- Transpose
- Transpose
- Transpose

Deploying self-attention to MCUs is challenging

- Typically many parameters
- Quantization of Softmax
- No efficient open-source kernels

In this work, we

- Developed 8-Bit self-attention kernels
- Implemented a tiny transformer on MCU
- Demonstrate an end-to-end use case

---

**Efficient Self-Attention Kernels**

- Transpositions and softmax are major bottlenecks in self-attention
- 73% of inference latency in SpAtten [2]
- Transpositions are impossible to parallelize on MCUs
- Softmax activation is sequence-dependent

First key idea: Fully quantize everything to 8 Bits

- Including activations & softmax
- Leverage SIMD instructions
- 8 Bit quantization doesn’t impact accuracy [3]

Second key idea: Introduce set of specialized kernels

- Merge softmax with matmul kernel, use quantized softmax activation [3]
- Merge transpositions with linear layer kernels by transposing data access on input matrices
- Parallelize over head dimension

Third key idea: Parallelize over appropriate dimensions

- Instruction parallelism: innermost loop ➞ Vector-products
- Thread parallelism: outermost loop ➞ Heads

---

**TinyRadar Transformer**

Modified version of CNN-TCN network [4] for gesture recognition

- Replaced dense convolutions by depthwise-separable convolutions
- Replaced TCN-layers by ViT-style Transformer encoder
- Downsamples the input data by a factor of 2x

---

**References**

[1]: Z. Dai et al., "CoAtNet: Marrying Convolution and Attention for All Data Sizes"
[2]: H. Wang et al., "SpAtten: Efficient Sparse Attention Architecture with Cascade Token and Head Pruning"
[3]: Kim et al., "1-BERT: Integer-only BERT Quantization"
[4]: M. Scherer et al., "TinyRadarNN: Combining Spatial and Temporal Convolutional Neural Networks for Embedded Gesture Recognition"
[5]: L. Li et al., "CMSIS-NN: Efficient Neural Network kernels for ARM Cortex-M CPUs"
[6]: A. Garofalo et al., "PULP-NN: Accelerating Quantized Neural Networks on Parallel Ultra-Low-Power RISC-V Processors"

---

**Conclusion**

In this work we presented

- Self-attention kernels with performance on-par with convolutions
- Close-to-linear multicore scaling
- A tiny Transformer that outperforms traditional CNN/TCNs
- End-to-end results on a real-world dataset, showing
  - 3.5% increase in Accuracy
  - 9.6x decrease in Latency
  - 9.6x decrease in Energy per Inference

The authors would like to thanks ArmaSuisse for funding this research